corresponding lock handle by linking the lock handle back on handle-free list 200. A reader attempting to use data structure 210 during lock handle disassociation will encounter either the next entry in handle-free list 200 or a pointer to data structure 210. Free list 200 entries are distinguished from internal data structure pointers by examining a bit alignment of the returned value. In this implementation, internal data structure pointers are aligned on a 4-byte boundary by setting their least significant two bits to zero, whereas handle-free list entries have their least significant pointer bit set to one (the associated lock handle index is shifted left one position to compensate). Therefore, a reader encountering a value with the least significant bit set knows that the associated lock handle does not correspond to internal data structure 210.

DEPR:
A reader encountering data structure 210 locks data structure 210 and checks a flag word embedded in data structure 210. When data structure 210 is prepared for deallocation, a "DEAD" flag bit is set therein, and data structure 210 is placed on a "pending deallocation" list of data structures. The DEAD flag bit is set under protection of a per-data structure lock.

DEPR:
The reader encounter data structure 210 with the DEAD flag bit set informs its controlling process that data structure 210 is pending deallocation. In this implementation, a lookup routine uses the existing per-data structure lock upon sensing the DEAD bit set, releases the lock, and informs its controlling process that the lock handle is no longer associated with an active internal data structure.

DEPV:
if the bit in rcc.sub.-- needctxtmask corresponding to the current processor is not set, return (do not execute the following steps);

DEPV:
if the bit in rcc.sub.-- needctxtmask associated with the current processor is already cleared, release rcc.sub.-- mutex and return;

DEPV:
clear the bit in rcc.sub.-- needctxtmask associated with the current processor to indicate that the current processor has completed the current generation;

DEPV:
if any bit in rcc.sub.-- needctxtmask is still set, release rcc.sub.-- mutex and return;

DEPV:
if rcc.sub.-- needctxtmask has a bit set (current generation not complete) or if rcc.sub.-- maxgen is less than rcc.sub.-- curgen (specified generation complete), return; and

End of Result Set

Generate Collection

L3: Entry 1 of 1 File: USPT Mar 10, 1998

DOCUMENT-IDENTIFIER: US 5727209 A
TITLE: Apparatus and method for achieving reduced overhead mutual-exclusion and maintaining coherency in a multiprocessor system utilizing execution history and thread monitoring

DEPR:
A dense per-thread bitmap data structure has an array of bits with one bit per thread. When a thread passes through a quiescent state since the data structure was reset, the corresponding bit is cleared.

DEPR:
A distributed per thread bitmap data structure embeds thread bits into a structure that facilitates thread creation and destruction. For example, a flag is used to track each thread in the data structure.

DEPR:
A hierarchical per-thread bitmap is a data structure that maintains a hierarchy of bits. The lowest level maintains one bit per thread. The next level up maintains one bit per group of threads and so on. All bits are preset to a predetermined state, for example, a one-state. When a thread is sensed in a quiescent state, its associated bit is set to a zero-state in the lowest level of the hierarchy. If all bits corresponding to threads in the same group are in a zero-state, the associated group bit in the next higher level is set to a zero-state and so on until either the top of the hierarchy or a non zero bit is encountered. This data structure efficiently tracks large numbers of threads. Massively parallel shared-memory multiprocessors should use a bitmap hierarchy mirroring their bus hierarchy.

DEPR:
Each bit is explicitly set to a predetermined state, preferably a one-state, by a reset signal 114.

DEPR:
Each bit (or group of bits for hierarchial bitmaps) has an associated generation counter. A global generation counter is incremented to reset summary of thread activity 106. When a thread is sensed in a quiescent state, and its associated bit is currently in a zero-state, its associated generation counter is compared with the global generation counter. If the counters differ, all bits associated with the quiescent thread are set to a one-state and the associated generation counter is set to equal the global counter.

DEPR:
A thread counter (not shown) may be used to indicate the number of threads remaining to be sensed in a quiescent state since the last reset signal 114. The thread counter is preset to the number of threads (or for hierarchial schemes, the number of threads in a group) and is decremented each time a thread bit is cleared. When the counter reaches zero, all threads have been sensed in a quiescent state since the last reset. If threads can be created and destroyed, the counters corresponding to the threads being destroyed must be decremented.

DEPR:
Callback processor 104 interfaces with the quiescence-indicating scheme chosen for summary of thread activity 106 and therefore has various possible implementations. For example, if the quiescence-indicating bits in summary of thread activity 106 have other purposes, no additional overhead need be

incurred by callback processor 104 in checking them. Consider a data structure having a dedicated summary of thread activity and a per-thread bit for indicating the occurrence of some unusual condition. Any thread accessing the data structure must execute special-case steps in response to the per-thread bit such as recording its quiescence before accessing the data structure.

DEPR:
readers just before or just after accessing the data structure protected by mutual-exclusion mechanism 90 (invoking callback processor 104 just after accessing the data structure will incur overhead unless the quiescence-indicating bits in summary of thread activity 106 have multiple purposes);

DEPR:
A callback processor 124 includes a one-bit-per processor bitmask. Each bit indicates whether its associated processor 16 must be sensed in a quiescent state before the current generation can end. Each bit corresponds to a currently functioning processor and is set at the beginning of each generation. When each processor 16 senses the beginning of a new generation, its associated per-processor context switch counter 122 value is saved. As soon as the current value differs from the saved value, the associated bitmask bit is cleared indicating that the associated processor 16 is ready for the next generation.

DEPR:
Callback processor 124 is preferably invoked by a periodic scheduling interrupt 126. Processor 16 may alternatively clear its bitmask bit if scheduling interrupt 126 is in response to an idle loop, a user-level process execution, or a processor 16 being placed off line. The latter case is necessary to prevent an off line processor from stalling the callback mechanism and causing a deadlock.

DEPR:
When all bits in the bitmask are cleared, callback processor 124 processes all callbacks 128 in a global current generation 130 and all callbacks 128 associated with the current processor in a per-processor current generation 131.

DEPR:
rcc.sub.-- olmsk is a bitmask in which each bit indicates whether a corresponding processor 16 is on line;

DEPR:
rcc.sub.-- needctxtmask is a field implementing summary of execution history 138 (the field is a bitmask in which each bit indicates whether a corresponding processor 16 has been sensed in a quiescent state);

DEPR:
Callback processor 124 is implemented by a function referred to as rc.sub.-- chk.sub.-- callbacks. Callback processor 124 is invoked by interrupt 126, which is preferably a hardware scheduling clock interrupt referred to as hardclock(), but only if one or more of the following conditions are met: rclocknxtlist is not empty and rclockcurlist is empty (indicating that the current processor is tracking a generation of callbacks and there are callbacks ready to join a next generation); rclockcurlist is not empty and the corresponding generation has completed; or the bit in rcc.sub.-- needctxtmask that is associated with the current processor is set. The latter condition ensures that if there is a current generation of callbacks, the current processor must be in, or have passed through, a quiescent state before the generation can end.

DEPR:
Handle table 170 accommodates NENTRIES 3 lock handles. A computer, such as computer 10, typically has a 32-bit (2 32) wide address bus which is sufficient to address 1024 NENTRIES (1024 3=2 30). Therefore, handle table 170 is adequately sized for all practical applications.

DEPR:
To deallocate data structure 210, it is first disassociated with its